# 北京化工大学

# 深度学习课程报告

**题 目：** 小样本条件下基于条件变分自编码器的数据生成方法研究


**班级：** 机研 2302

**学号：** 2023200723

**姓名：** 叶立锴


2024 年 1 月 5 日

# 第 1 章 绪论

在重大装备实际运行过程中，由于故障样本数量较少，故障诊断领域也面临着小样本问题的挑战。小样本问题在故障诊断中尤为突出，因此，如何有效地利用有限的故障数据进行准确的诊断成为了故障诊断领域中一个重要的研究方向。

目前，故障诊断技术已经发展到多种不同的层次，包括利用数据分析、数据处理、数据挖掘以及利用人工智能技术。

深度学习理论由 Hinton 等人引入科学界，这也开启了一个全新的智能化时代。随着时间的推移，深度学习的学习算法不断进步，它的学习特征、模型结构、算法规则以及模型之间的相互作用，已经被广泛用于各个行业，并且受到越来越多的关注。作为一种新兴的机器学习技术，深度学习具有许多独特的特点和优势。其中最突出的三个特点是：自适应特征提取能力、大规模数据处理能力和其复杂的网络结构。深度学习在设备故障诊断中具有很大的优势和潜力。在未来的研究中，需要进一步探索深度学习在设备故障诊断中的应用，完善深度学习算法，提高故障诊断的准确性和可靠性。

在实际生产过程中只能获得非常少的故障数据，这就导致了小样本问题的出现。如果不能有效地利用这些少量的故障数据，将会极大地限制对机器故障的诊断能力和准确性，这也成为了困扰人们的新问题，要解决这个问题就必须生成新的故障数据。

因此，本课题研究验证了基于条件变分自编码器的数据生成方法，构建了条件变分自编码器（Conditional Variational Auto-Encoder, CVAE）数据生成网络模型；最后，基于相似性和多样性，对重构信号进行了质量检验，并对生成信号的故障信息进行了验证，得到的新生成的故障数据可以提高故障信息分类的准确率，对于解决小样本问题具有重要的意义。

# 第 2 章 基于分量信号的 CVAE 数据生成网络构建

## 2.1 变分自编码器

### 2.1.1 变分自编码器网络结构

VAE（Variational Autoencoder）是一种基于神经网络的无监督学习方法，其核心思想是学习数据的潜在分布，并通过采样生成新的数据。VAE 通过对输入数据的编码和解码，来间接地学习出隐变量的分布。具体来说，VAE 的编码器将真实数据映射为一个潜在空间中的分布参数，而解码器则将潜在变量映射回到原始数据空间中。在这个过程中，VAE 还引入了一个重参数化技巧，使得可以对潜在变量进行采样，从而更好地训练模型。VAE 的基本结构包括编码器和解码器两个部分。VAE 结构如下图 2-1：

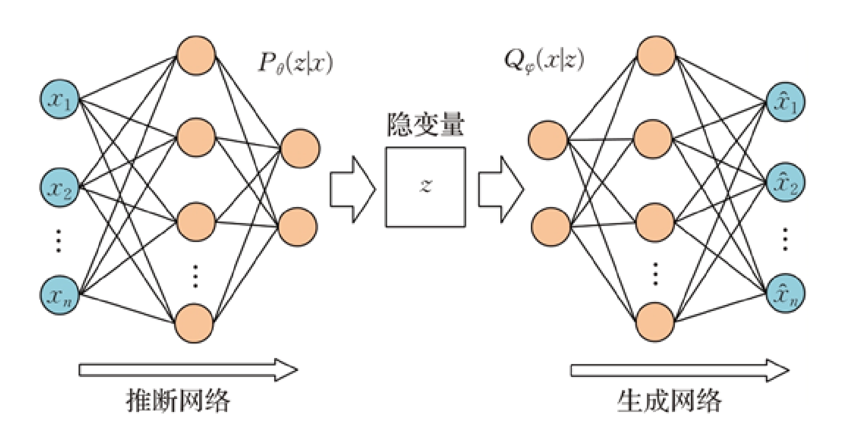



图 2-1 VAE 网络结构图

但其关注的不仅是重构误差，还包括用于生成数据的隐变量，通过采样生成新的数据。通过贝叶斯公式，可以推断出隐变量 $Z$ 的概率分布，公式 3-1 如下：

$$P(Z|X)=\frac{P(X|Z)P(Z)}{\int P(X|Z)P(Z)dZ}$$

其中 $X$ 为训练数据集，$Z$ 为隐含变量，$P(Z|X)$为已知 $X$ 是 $Z$ 的概率分布。

采用损失函数作为指标，可以有效地比较生成的样本和真实样本的误差，以此作为基础，利用反向传播算法对模型进行优化，以提高预测准确性。

VAE 的损失函数由两个部分组成：重建损失和 KL 散度损失。前者用来比较模型预测的结果与实际数据之间的差异，而后者则用来比较模型预测的结果与实际数据之间的差异。

重构损失是指通过解码器生成的样本与原始样本之间的误差，可以使用均方误差或交叉熵来计算。KL 散度可以衡量两个分布之间的相似度，当两个分布接近时，KL 散度接近于 0，当两个分布差异越大时，KL 散度越大[33]。因此，VAE 的损失函数可以写作重构损失与 KL 散度损失的加权和，其中权重可以通过超参数调整。通过最小化这个损失函数，从而学习到生成样本的概率分布，生成更加真实的样本。损失函数公式如下：

$$L_{\mathrm{oss?}} = K_{\mathrm{L}}\left(N\left(\mu,\sigma^2\right) \| N(0,1)\right) + \| X - \hat{X} \|^2$$

右侧第一项为隐变量分布与标准正态分布间的 KL 散度；右侧第二项为原始数据 $X$ 和生成数据间的重构损失。

由于 KL 散度的损失，生成的样本很可能与真实样本的分布存在较大的差异，从而使得它们很难被准确地识别出来。因此，训练过程中，应该尽量保证样本的多样性，以便获得更加丰富的结果。

故本文选取有特定条件的变分自编码器，它是 VAE 的改进版。

### 2.1.2 变分自编码器特点

VAE 是一种无监督学习方法，可以生成具有多样性的样本，并且可以学习到数据的潜在结构，进行数据压缩和降维。它可以通过调整潜在空间中的向量来生成具有特定属性的样本，广泛应用于图像生成、图像编辑等领域。同时，VAE 可以控制生成样本的多样性和质量，容易优化和调整，并且可以与其他深度学习模型结合使用，如 GAN，以获得更好的生成效果。

## 2.2 条件变分自编码器

### 2.2.1 条件变分自编码器网络结构

CVAE 是一种基于深度学习的生成模型，它是 VAE 的一种扩展形式，可以处理具有条件属性的数据。其核心思想是使用编码器将输入数据和条件信息一起映

射到潜在空间中的分布，然后使用解码器将潜在空间中的采样和条件信息一起映射回输入空间。其结构如下图 2-2 所示：

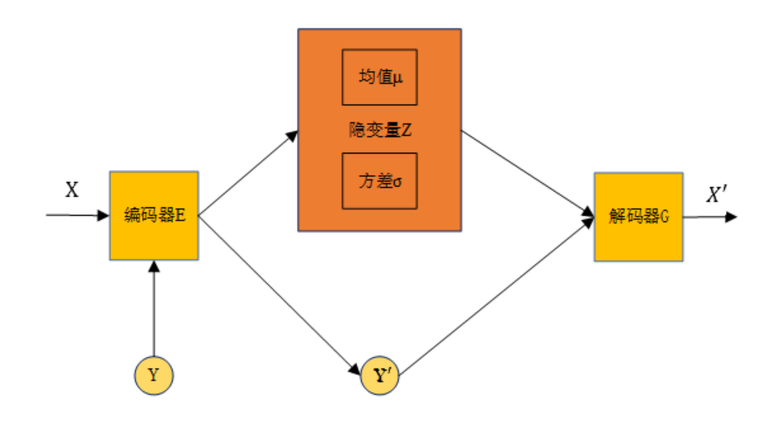



**图 2-2** CVAE 网络结构图

CVAE 的结构和 VAE 类似，但是在编码器和解码器中加入了条件信息。其中，不同点在于 CVAE 的编码器的输入是数据和条件信息，输出是潜在空间中的分布参数，解码器的输入是潜在空间中的采样和条件信息，输出是重构的数据。在训练时，CVAE 网络可以学习到数据和条件信息之间的复杂关系，在生成数据时，可以通过调整条件变量的值来控制生成数据的特征。

此外，二者的损失函数也有不同，在 VAE 中，损失函数只包括重构误差和 KL 散度，而在 CVAE 中，则还包括条件熵，这样保证了生成样本的多样性。

条件熵是指在给定某个条件下，随机变量的不确定性量度。具体来说，如果有一个随机变量 $X$ 和条件变量 $Y$，则条件熵 $H(X|Y)$表示在已知 $Y$ 的情况下，$X$ 的不确定性。其公式如下：

$$H(X|Y) = -\Sigma p(x,y) * \log(p(x|y))$$

其中，$p(x,y)$ 表示 $X$ 和 $Y$ 的联合分布，$p(x|y)$ 表示在已知 $Y$ 的情况下，$X$ 的条件分布，log 表示以 2 为底的对数。

在实际应用中，可以通过调整模型的超参数、损失函数权重等方式来控制条件熵的大小。通常可以尝试不同的参数组合，并通过交叉验证等方法来选择最优的参数。因此，在设计 CVAE 模型时需要平衡条件熵和其他损失函数之间的关系，以达到最优的生成效果。

### 2.2.2 条件变分自编码器特点

本文的目的是扩充故障样本数据，正是由于 CVAE 可以通过控制输入的标签信息来生成不同类型的样本，可以生成特定标签下的样本，使得生成的样本更加具有针对性。对于分量信号比较多、信号比较复杂的情况，CVAE 可以通过输入数据的标签信息来生成不同类型的样本，从而增加生成样本的多样性，更好地扩充故障样本数据。

## 2.3 基于一维振动信号的 CVAE 网络构建

本论文中采用的 CVAE 模型是基于 Python 编程软件中的 Keras 模块实现的。其中，一维分解方法对应的编码器由两个卷积层和三个全连接层构成，解码器由三个全连接层和两个卷积层构成。其结构图 2-3 如下：

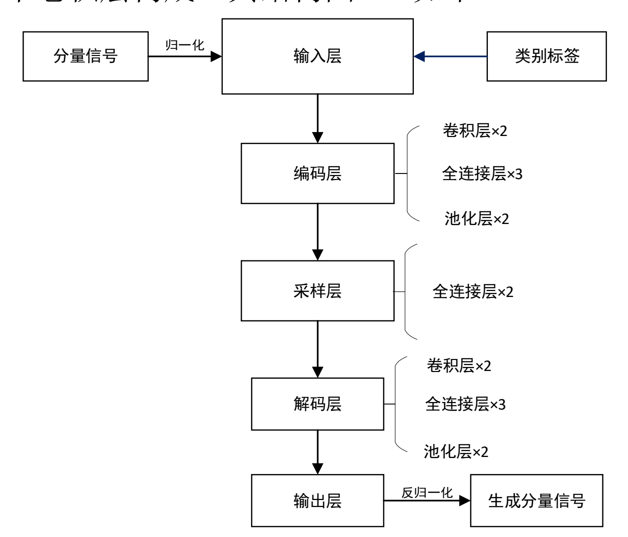



**图 2-3** 一维分量对应 CVAE 网络结构

本模型将独热编码作为类别标签对分量信号进行训练。独热编码是一种将离散变量转换为连续变量的方法，可以将离散的标签信息转换为一个向量，向量的长度等于标签的种类数，每个维度对应一个标签，其中只有一个维度的值为 1，其余维度的值为 0，表示该样本的标签是哪一个。在该模型中，独热编码作为标签，来控制生成的样本的类别，从而保证生成样本的质量。

2.3.1 输入层

首先对分量信号数据的行和列分别进行归一化处理，主要目的是加速网络的收敛和提高网络的稳定性，同时也可以提高网络的泛化能力和避免梯度消失或梯度爆炸等问题。其归一化公式 3-4 如下：

$$x_{norm}=\frac{x-x_{\min}}{x_{\max}-x_{\min}} \quad (3\text{-}1)$$

其中，$x$ 表示原始数据，$x_{\min}$ 和 $x_{\max}$ 分别表示数据集中的最小值和最大值。$x_{norm}$ 表示归一化后的数据。

2.3.2 编码层

在编码器结构中，编码器的第一个和第二个卷积层都使用了 64 个大小为 25 的卷积核。两个卷积层的池化窗口尺度都相等，都有 4 个，这种编码方法可以有效地提高数据的准确性和可靠性。然后随着 Flatten 层的加载，三维张量变换成一维向量，而两个全连接层则被划分为 200 维与 100 维，以便将输入数据映射到潜在空间中。其参数如表 2-1 所示：

**表 2-1** 编码层参数

| | 参数 | 数值 |
|---|---|---|
| 第一个卷积层 | 卷积核数量 | 64 |
| | 卷积核大小 | 25 |
| 第二个卷积层 | 卷积核数量 | 64 |
| | 卷积核大小 | 25 |
| 池化层 | 数量 | 2 |
| | 窗口大小 | 4 |
| 第一个全连接层 | 维度 | 200 |
| 第二个全连接层 | 维度 | 100 |

2.3.3 采样层

采样层由两个全连接层构成，通过两个全连接层计算得到的潜在变量的均值和方差，然后由 Lambda 函数实现，其结构包括潜在变量的均值、方差两个输入，以及一个输出潜在变量 Z。Lambda 函数内部生成一个标准正态分布的随机数，然后将其与潜在变量的均值和方差计算，得到的值结合起来，通过乘以标准差并加上均值得到最终的潜在变量 Z。

### 2.3.4 解码层

其解码器的第一个全连接层和编码器的最后一个全连接层的维度相同，都是200个。然后是三个全连接层，维度分别是400、800和1600，用于将潜在空间的向量解码成原始信号数据。最后两个卷积层的大小和卷积核个数与编码器的卷积层相同，都是64个大小为25的卷积核。其参数如下表2-2：

**表 2-2** 编码层参数

| | 参数 | 数值 |
|---|---|---|
| 第一个全连接层 | 维度 | 200 |
| 第二个全连接层 | 维度 | 400 |
| 第三个全连接层 | 维度 | 800 |
| 第四个全连接层 | 维度 | 1600 |
| 第一个卷积层 | 卷积核数量 | 64 |
| | 卷积核大小 | 25 |
| 第二个卷积层 | 卷积核数量 | 64 |
| | 卷积核大小 | 25 |

### 2.3.4 输出层

输出层通过对网络模型中的数据进行编码和解码，然后对其进行反归一化，最终将处理结果存储起来。首先将生成数据从一维数组变为与输入数据形状形同的二维数组。接下来，对于每一列、每一行依次进行反归一化处理，得到反归一化后的数据。其反归一化公式如下：

$$x' = x_{norm} \times \left(x_{\max} - x_{\min}\right) + x_{\min}$$

其中，$x_{norm}$表示归一化后的数据，$x_{\min}$和$x_{\max}$分别表示数据集中的最小值和最大值。$x'$表示反归一化后的数据。

最终将处理后的数据保存到 mat 文件中。

## 2.4 基于二维振动信号的 CVAE 网络构建

与一维分量不同，二维分量信号数据集为四维矩阵，数据大小不同，故对应的 CVAE 网络也有着不同，其结构如下图 3-4：

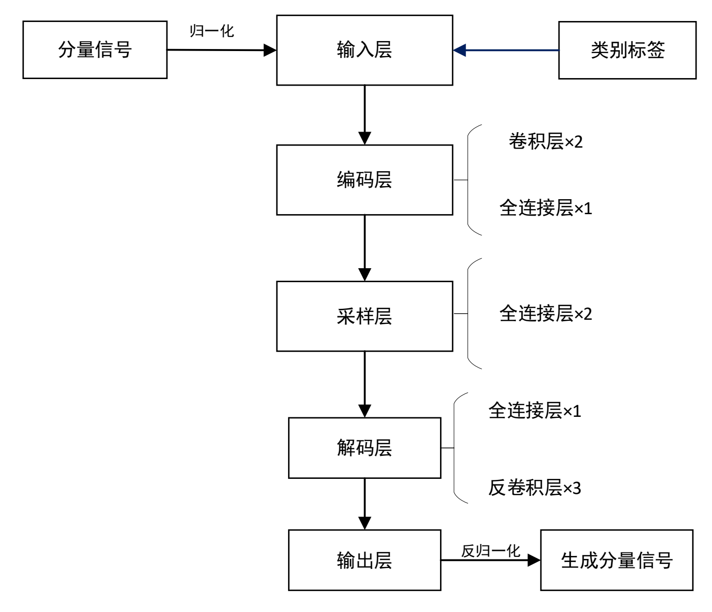



**图 3-4** 二维分量对应 CVAE 网络结构

### 2.4.1 输入层

二维分量的数据集为四维数据，输入层建立一个三维矩阵，将四个分量数据集的同一分量提取出来，将输入数据转换为可供网络模型处理的标准化数据格式。然后通过循环将这四个数据集中的二维分量数据合并成一个二维数组，并使用归一化将每个特征的数值范围缩放到 0 到 1 之间，与一维分量的归一化公式相同。归一化后，记录每个特征的最小值和最大值，以便后续预测时进行逆归一化。最后，将归一化后的数据重新变成三维数组。同时，将对应的独热码作为类别标签。在训练过程中，将每个数据样本的标签作为输入的一部分，与二维图谱数据一起输入到神经网络中。

### 2.4.2 编码层

编码器的结构由两个卷积层 Conv2D 和一个全连接层 Dense 组成。第一个卷积层的作用是将输入的二维分量数据进行特征提取，提取出数据中的重要特征。第二个卷积层进一步提取特征，并将特征的维度降低，从而减少模型的参数量。然后通过一个全连接层将特征展平成一维向量，并将其映射到一个低维向量，表示数据的潜在表示。这个低维向量的维度是通过参数 Latent dim 来指定的。Relu 激活函数被广泛应用于编码器的各个部分，特别是卷积层和全连接层，它们的应

用可以显著提升模型的非线性处理能力。最终，编码器的输出是一个大小为 Latent dim 的低维向量，表示输入数据的潜在表示。其具体参数如下表 3-3：

**表 3-3** 编码层参数

| | 参数 | 数值 |
|---|---|---|
| 第一个卷积层 | 卷积核数量 | 8 |
| | 卷积核大小 | 9 |
| | 激活函数 | Relu |
| 第二个卷积层 | 卷积核数量 | 16 |
| | 卷积核大小 | 9 |
| | 激活函数 | Relu |
| 全连接层 | 维度 | 16 |
| | 激活函数 | Relu |
| Latent dim | 维度 | 3 |

### 2.4.3 采样层

采样层由两个完整的连接层组成，它们可以用来计算潜在变量的平均值和方差，并使用 Lambda 函数来处理这些信息。Lambda 函数内部生成一个标准正态分布的随机数，然后将其与潜在变量的均值和方差计算，得到的值结合起来，通过乘以标准差并加上均值得到最终的潜在变量 Z。

### 2.4.4 解码层

解码层包括一个全连接层、三个反卷积层。首先，潜在变量 Z 经过一个全连接层，将潜在变量表示为上一层卷积层输出的形状。然后，经过两个反卷积层，将其转换为与上一层卷积层输入相同的形状，并逐渐减小通道数。最后，经过一个输出通道数为 1 的反卷积层，输出训练后的数据。最后一层的激活函数为 sigmoid，它的作用是将所有的实数值映射到[0,1]的范围内。这样，解码器的输出就与输入数据的范围相同，方便后续的比较和评估。其参数如表 2-4 所示：

**表 2-4** 解码层参数

| | 参数 | 数值 |
|---|---|---|
| 全连接层 | 维度 | 16 |
| | 激活函数 | Relu |
| 第一个反卷积层 | 卷积核数量 | 16 |
| | 卷积核大小 | 9 |
| | Kernel size | 3 |
| | 激活函数 | Relu |
| 第二个反卷积层 | 卷积核数量 | 8 |

| | | |
|---|---|---|
| | 卷积核大小 | 9 |
| | Kernel size | 3 |
| | 激活函数 | Relu |
| 第三个反卷积层 | 卷积核数量 | 1 |
| | 卷积核大小 | 9 |
| | Kernel size | 3 |
| | 激活函数 | sigmoid |

### 2.4.5 输出层

输出层就是将训练好的数据进行反归一化处理，与一维分量的反归一化处理方法相同。然后将输出数据进行一系列的调整，将其形状调整为与输入数据的形状相同。

## 2.5 CVAE 网络参数选取

模型建立之后，需要通过不断训练并更新模型参数。在进行训练之前，需要设置一系列参数。参数也需要不断调整，寻找最优值，使得网络模型的生成效果达到比较理想的程度。一维分量对应的 CVAE 网络模型的主要参数调整如下：

（1） 学习率

学习率在神经网络中起着至关重要的作用，它可以确定目标函数是否可以达到最佳状态，并且可以预测收敛所需的时间[34]。

学习率的调整从较小值逐渐增大，观察损失值的变化。调整区间为 0.0001-0.01，可以观察到刚开始的准确率较低且不稳定，随后逐渐增大，达到一定程度后再次下降。当损失值开始收敛并且不再有太大的波动时，可以认为学习率的选择比较合适。

（2） epoch

epoch 的大小对于网络模型的性能有着重要的影响。如果 epoch 太小，网络无法充分地学习训练数据集中的特征，导致欠拟合；如果 epoch 太大，网络可能会过度拟合训练数据集，从而泛化能力下降。

（3） 隐变量

Latent dim 是指 CVAE 网络中的潜在空间维度。CVAE 网络模型的维度越高，模型表达能力越强，但计算也会更加复杂，可能会导致过拟合，导致泛化能力下

降。当维度较低时，模型的表达能力会受到影响，这将使得计算复杂度大大增加，从而使得模型无法有效地捕捉到数据中的关键信息，从而导致模型的性能大幅度下降。经过不断调整，最终将 Latent dim 大小确定为 3。

本文以分类质量检验和示例验证作为性能指标观察模型训练结果。其中，两类模型参数的设置如下表 2-5、2-6 所示：

**表 2-5** 一维分解对应 CVAE 模型参数汇总

| 参数 | 数值 |
|---|---|
| Num classes | 4 |
| Latent dim | 3 |
| Batch size | 40 |
| 迭代次数 epoch | 500 |
| 学习率 lr | 0. 001 |
| 卷积层层数 | 4 |
| 池化层层数 | 6 |
| 全连接层层数 | 4 |

**表 2-6** 变分二维分解对应 CVAE 模型参数汇总

| 参数 | 数值 |
|---|---|
| Num classes | 4 |
| Latent dim | 3 |
| Batch size | 30 |
| 迭代次数 epoch | 300 |
| 学习率 lr | 0. 001 |
| 卷积层层数 | 2 |
| 反卷积层层数 | 3 |
| 全连接层层数 | 3 |

# 第 3 章 基于生成信号的质量检验与案例验证

## 3.1 生成信号质量检验

### 3.1.1 t-SNE

t-SNE 是一种高效的非线性降维技术，能够将高维空间中的相似点映射到低维空间中的相邻点，而将高维空间中的不相似点映射到低维空间中的远离点[35]。

在数据可视化中，t-SNE 可以帮助识别数据中的聚类结构和相似性关系。本文采用的两种分解方法生成信号与原信号对应分布如图 4-19、图 4-20：

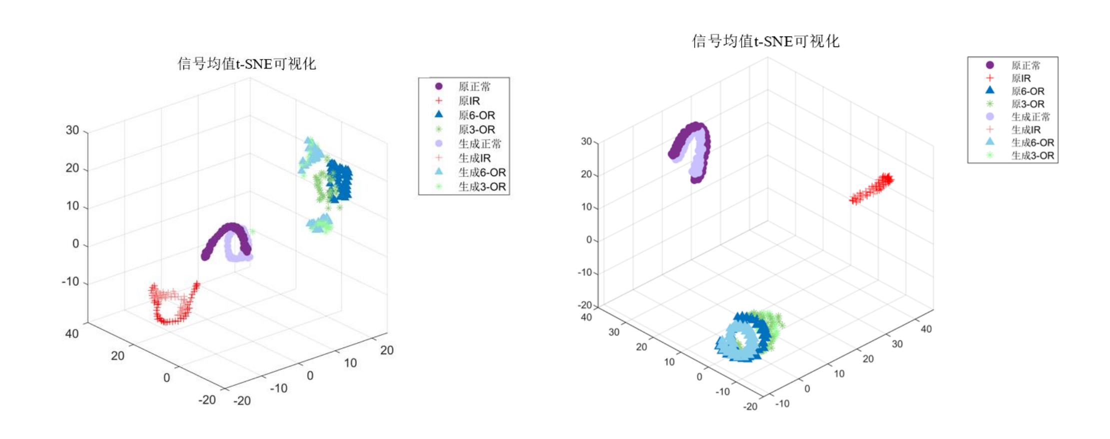



**图 3-1** 一维分解方法　　　　**图 3-2** 变分二维分解方法

以上两张图显而易见，变分二维分解的方法在 t-SNE 可视化中的效果好于一维分解方法。

### 3.1.2 余弦相似度

余弦相似度是一种用于计算两个向量之间相似度的方法，因此，余弦相似度检验可以用来评估生成信号与原始信号之间的相似度，从而判断生成信号的质量是否符合要求。在本论文中，可以将原始信号和生成信号看作两个向量，然后使用余弦相似度来计算它们之间的相似度。余弦相似度的计算公式如下：

$$\cos\theta = (A \cdot B) / (\| A \| \times \| B \|)$$

其中，$A$ 和 $B$ 分别代表两个向量，$\cdot$ 表示内积运算，$\| A \|$和$\| B \|$表示向量的模长。余弦相似度的取值范围在-1 到 1 之间，值越接近 1 表示两个向量越相似，值越接近-1 表示两个向量越不相似[36]。

在用余弦相似度检验原始信号与生成信号之间的相似性时，为了更直观的展现生成信号的效果，本文将原始信号与自身之间的余弦相似度和原始信号与生成信号之间的余弦相似度的图像放在了一张图里，这样清晰地看出生成信号的质量很好。如图 4-21、图 4-22 所示：

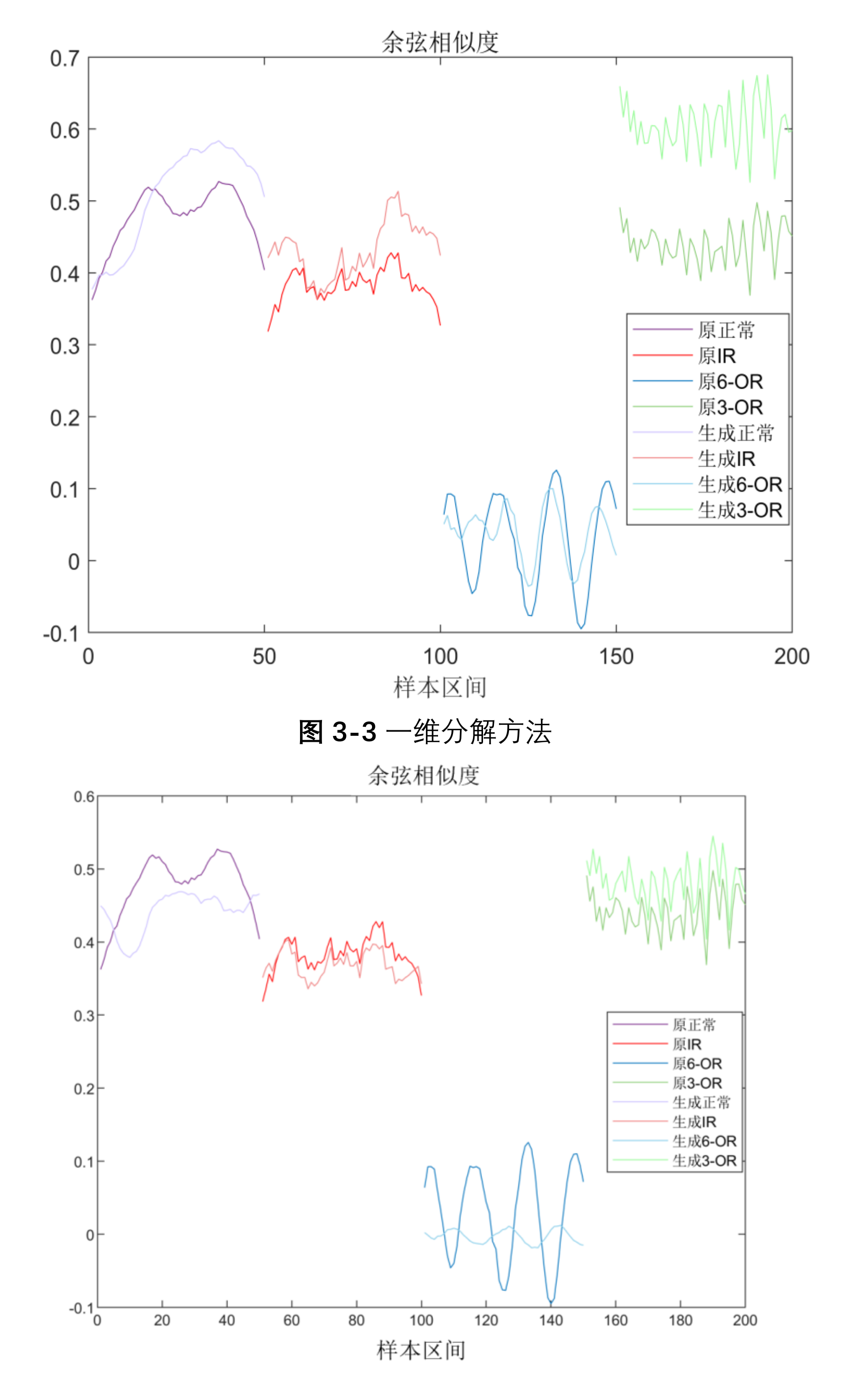



**图 3-3** 一维分解方法

**图 3-4** 变分二维分解方法

以上两张图显而易见，变分二维分解的方法在余弦相似度的验证中好于一维分解方法。

### 3.1.3 最大均值差异

MMD（Maximum Mean Discrepancy）是一种用于检验两个分布之间相似度的方法，可以用来检验本文的原始信号和生成信号之间的相似度。

MMD 的计算方法是通过比较两个分布在特征空间中的期望值之间的差异来判断它们之间的相似度。在本文中，可以将原始信号和生成信号分别看作两个分布，然后使用 MMD 来比较它们之间的相似度。MMD 的计算公式如下：

$$MMD(\mathrm{P},\mathrm{Q}) = \| \mathrm{E}(\mathrm{phi}(\mathrm{x})) - \mathrm{E}(\mathrm{phi}(\mathrm{y})) \|\hat{}\ 2$$

其中，P 和 Q 分别代表两个分布，phi(x)和 phi(y)分别表示将信号 x 和 y 映射到特征空间中的函数。|| • ||表示向量的模长，E( • )表示期望值。

MMD 的取值范围在 0 到正无穷之间，值越接近 0 表示两个分布越相似，反之则说明它们之间的差异越大。两种分解方法原信号与生成信号的最大均值差异如下图 4-23、图 4-24 所示：

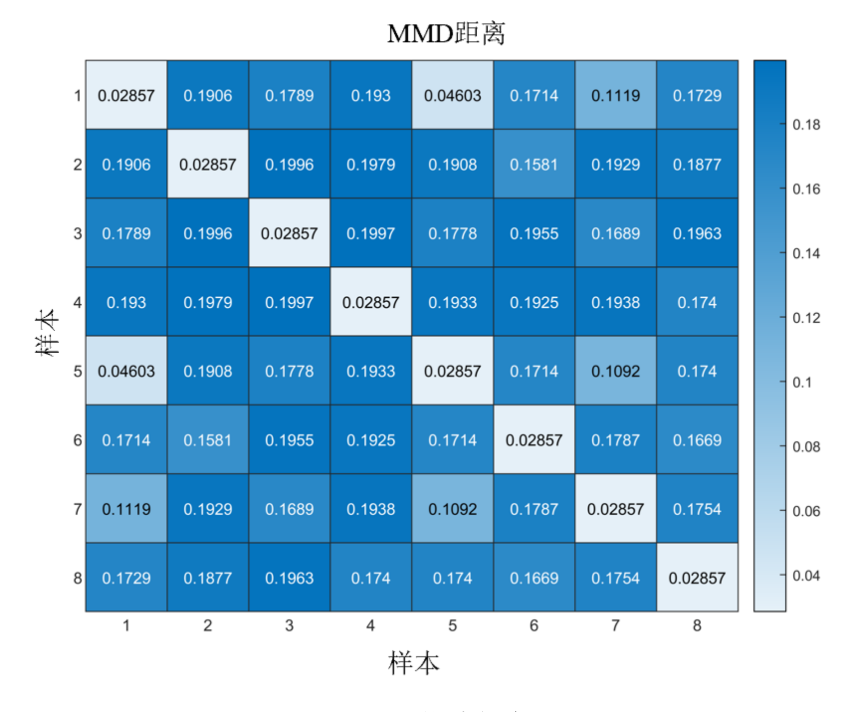



**图 3-4 一维分解方法**

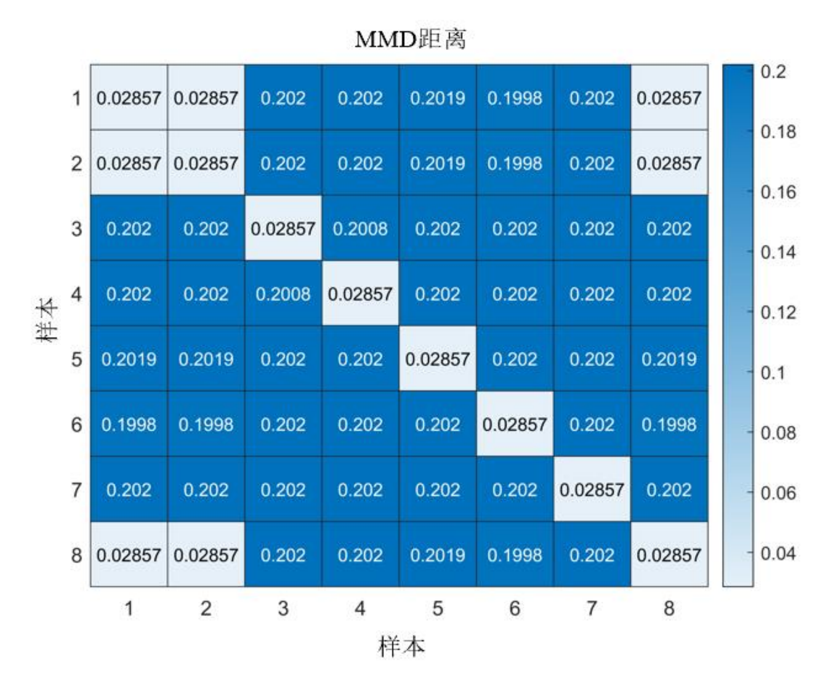



**图 3-5** 变分二维分解方法

上图横坐标 1-4 代表四个原始样本，5-8 代表生成的样本，由图可知，1 与 5 进行对比，2 与 6 进行对比，以此类推，可见通过最大均值差异来检验相似性，分解这种方法要略优于变分二维分解。

## 3.2 案例验证

### 3.2.1 小样本条件下故障信息分类准确率

在故障诊断领域中，由于样本数量较少，故障识别准确率可能会受到一定的影响。这个问题在实际应用中比较常见，因为故障数据的获取比较困难。同时，由于各种类型的故障数据的差异，以及“学习”中未曾接触的信息，使得模型的训练效果无法达到理想的水平，从而引发了过拟合、欠拟合以及其他各种误差，从而严重损害了故障识别的精度。

为了模拟实际过程中的小样本问题，分别将三类故障原始样本数量设置为 10、30、50，分别代入故障信息分类模型中，得到了三类状态信息分类的准确率如图 4-31、32、33：

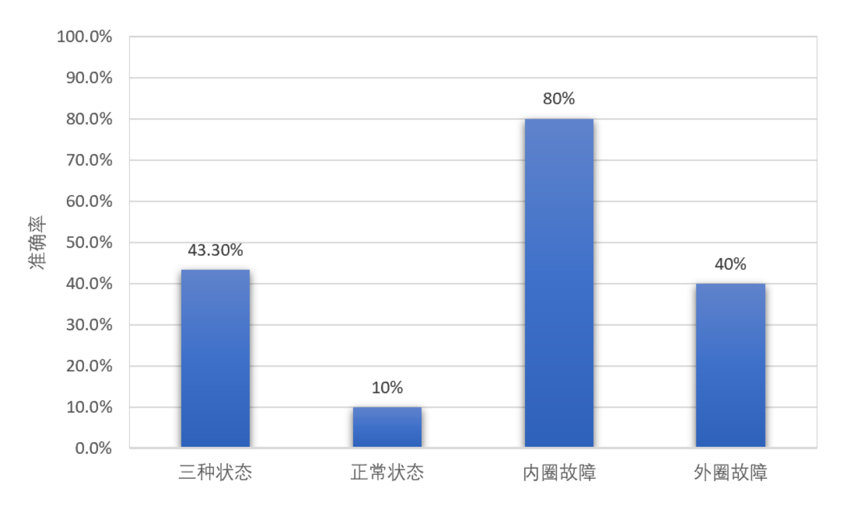



图 3-6 各类样本数量为 10 的条件下三类状态的故障信息分类准确率

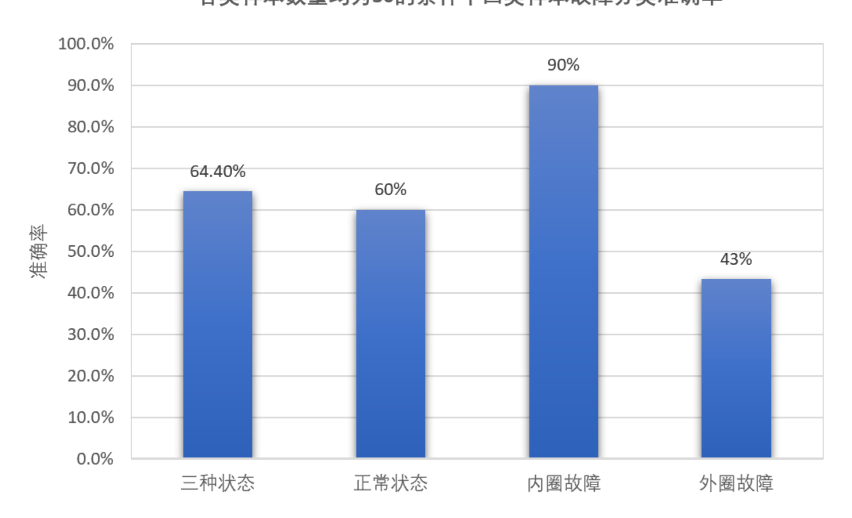



图 3-7 各类样本数量为 30 的条件下三类状态的故障信息分类准确率

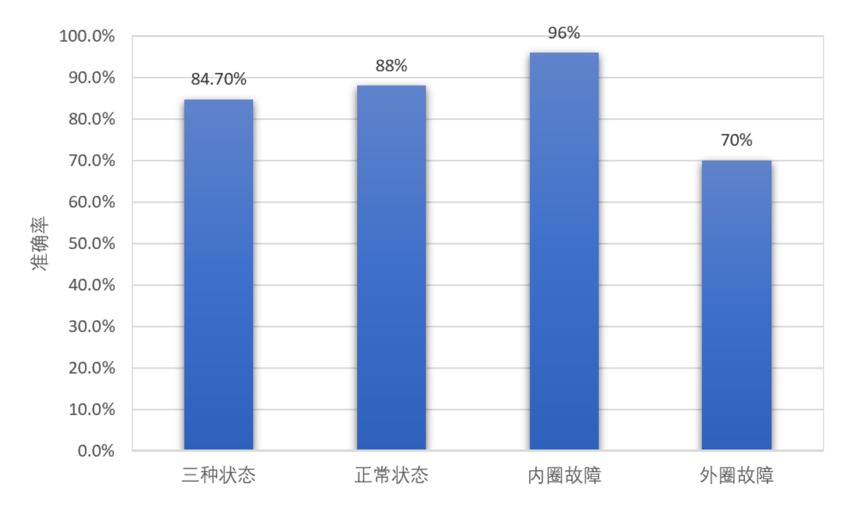



图 3-8 各类样本数量为 50 的条件下三类状态的故障信息分类准确率

由分类结果可知，样本数量越少，每种故障分类的效果就越不理想，整体三类故障的分类情况也不理想，通过利用全部的原始样本，最终将三类的故障分类准确率由 43.3%提高到了 84.7%，但这还是达不到实际工程理想的效果。

### 3.2.2 样本充足条件下故障信息分类准确率

为了验证本文生成的数据故障信息，本文将两种方法各生成的三类样本和原始样本作为输入样本，其中正常状态样本及其他两类样本数量均为 150，带入到故障信息分类的模型中，这样就模拟了样本充足的条件，得到了三类状态信息分类的准确率为 98.3%，各类状态信息具体分类的准确率如图 4-34：

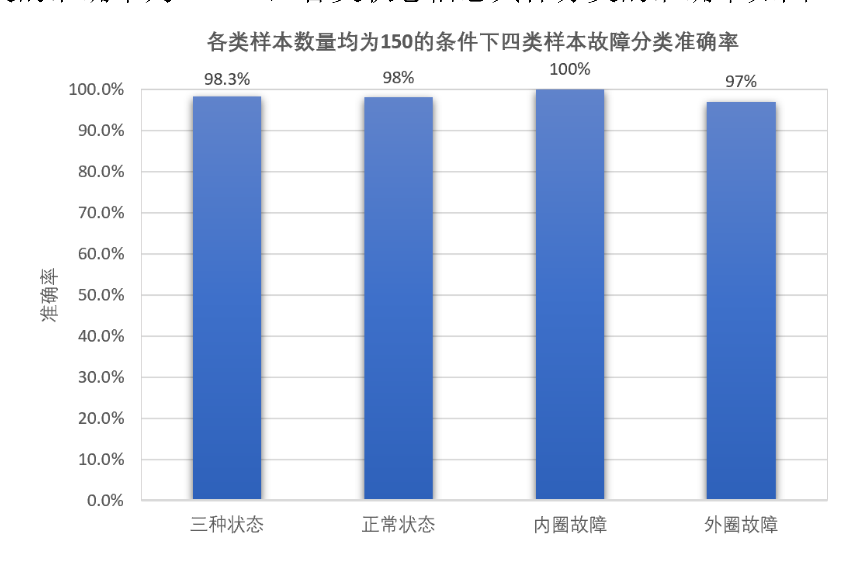



图 3-9 样本充足条件下四类样本的故障信息分类准确率

由图可知，两种方法得到的生成样本均能有效反映出原始样本中的故障信息，证明了本文的可行性。同时也反映了故障样本充足的条件下，故障信息分类的准确率也会大大提高，验证了本论文的 CVAE 网络模型的可靠性。